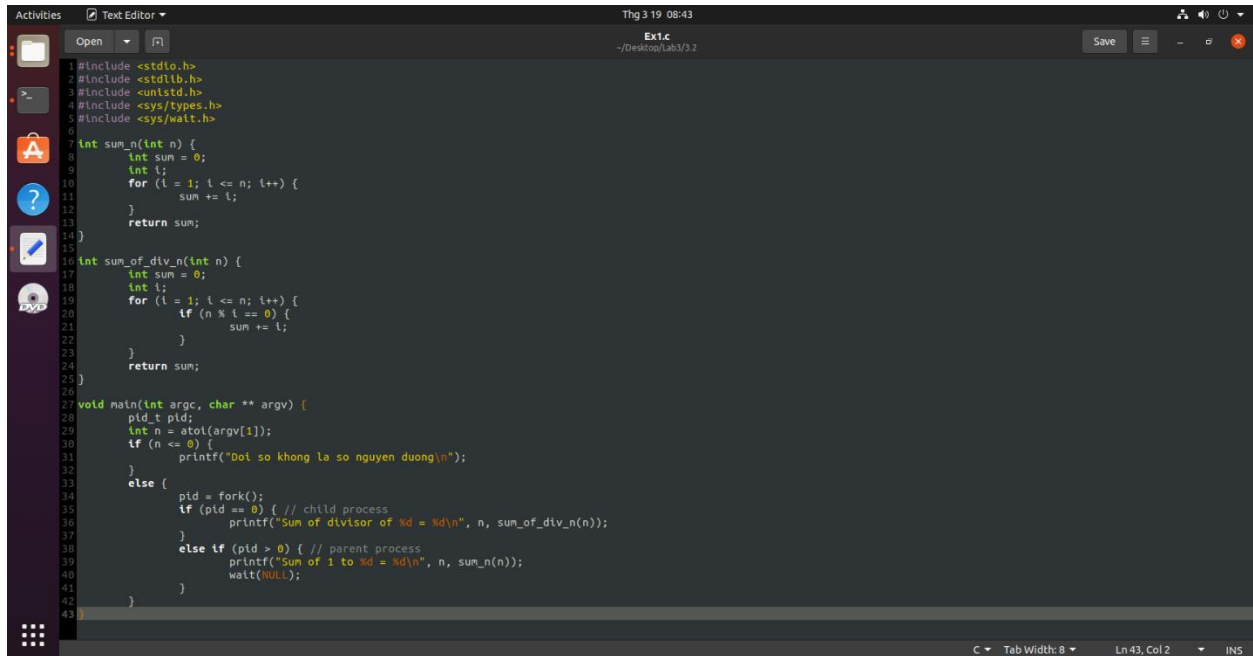


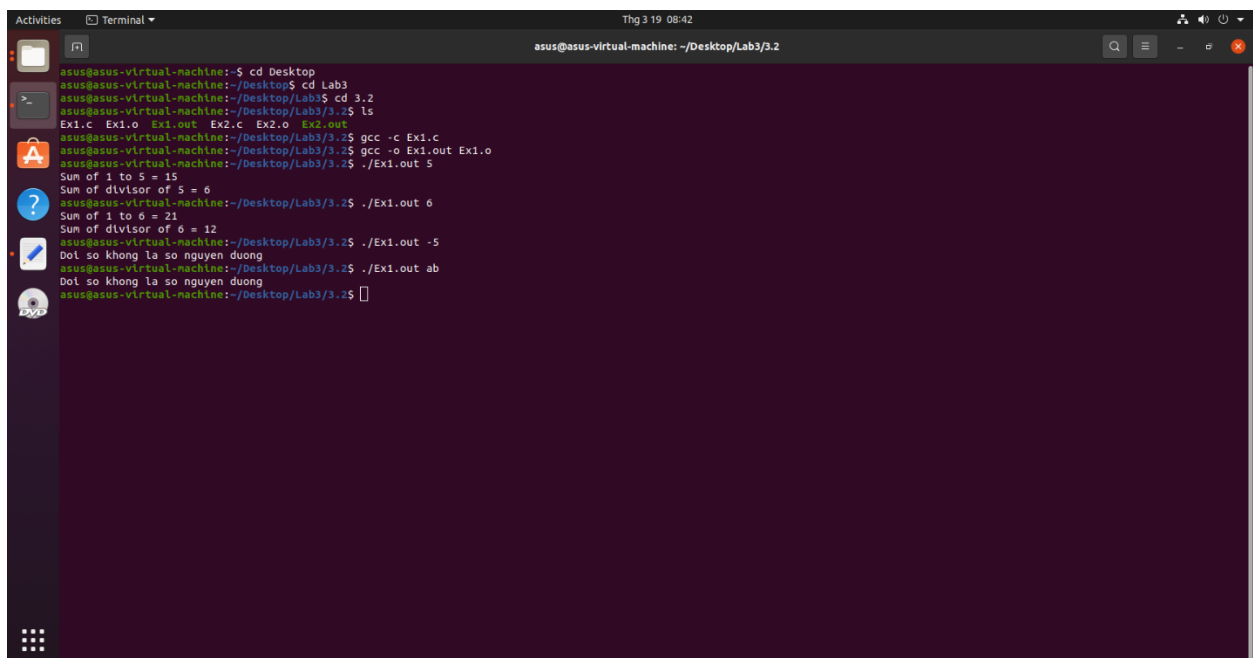
LAB 3

3.2 – Tạo tiến trình

Bài 1:

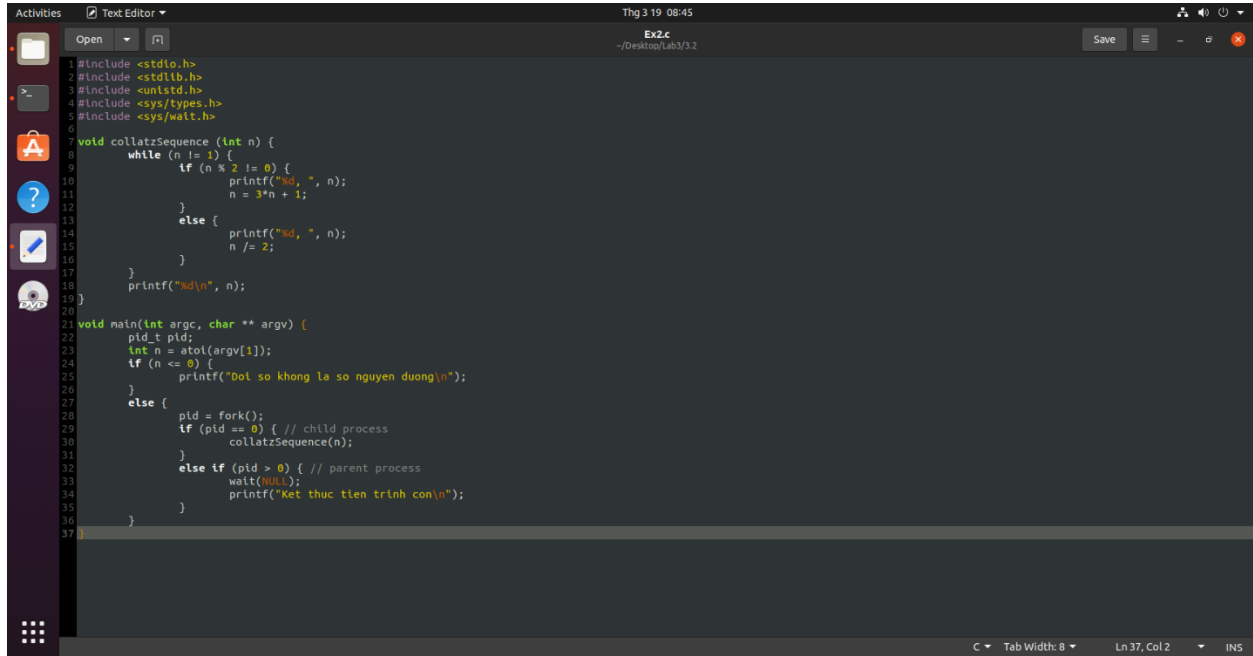


```
1#include <stdio.h>
2#include <stdlib.h>
3#include <unistd.h>
4#include <sys/types.h>
5#include <sys/wait.h>
6
7int sum_n(int n) {
8    int sum = 0;
9    int i;
10   for (i = 1; i <= n; i++) {
11       sum += i;
12   }
13   return sum;
14}
15
16int sum_of_div_n(int n) {
17   int sum = 0;
18   int i;
19   for (i = 1; i <= n; i++) {
20       if (n % i == 0) {
21           sum += i;
22       }
23   }
24   return sum;
25}
26
27void main(int argc, char ** argv) {
28   pid_t pid;
29   int n = atoi(argv[1]);
30   if (n <= 0) {
31       printf("Dot so khong la so nguyen duong\n");
32   }
33   else {
34       pid = fork();
35       if (pid == 0) { // child process
36           printf("Sum of divisor of %d = %d\n", n, sum_of_div_n(n));
37       }
38       else if (pid > 0) { // parent process
39           printf("Sum of 1 to %d = %d\n", n, sum_n(n));
40           wait(NULL);
41       }
42   }
43}
```

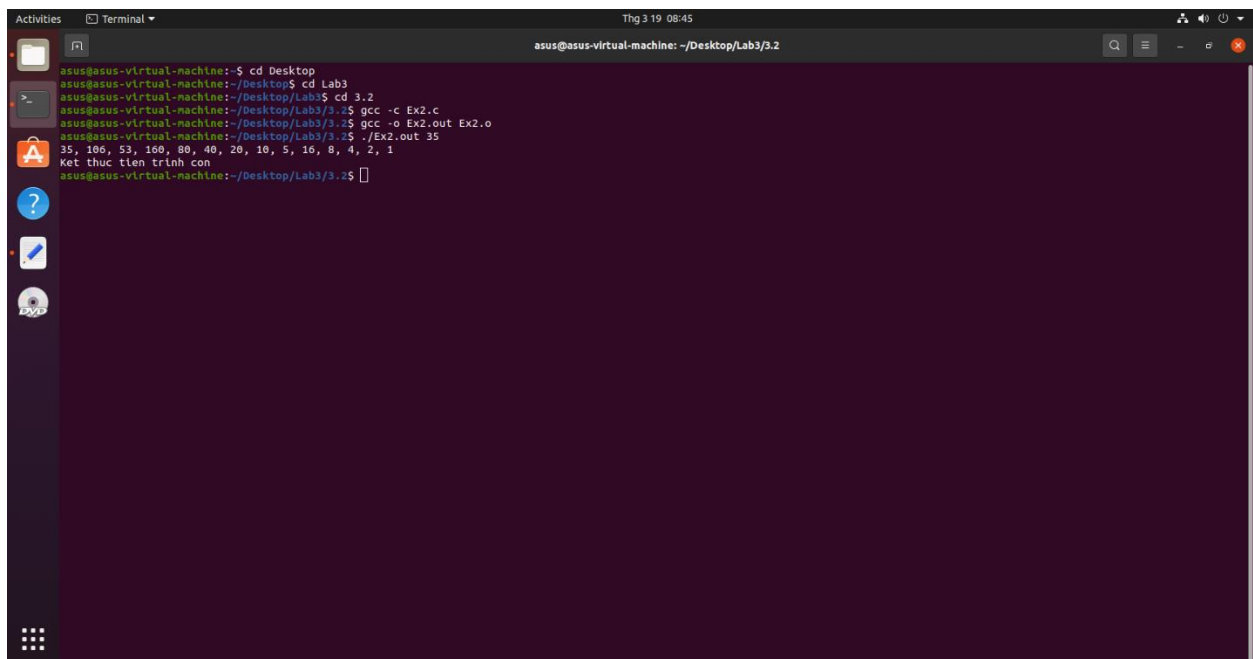


```
asus@asus-virtual-machine:~$ cd Desktop
asus@asus-virtual-machine:~/Desktop$ cd Lab3
asus@asus-virtual-machine:~/Desktop/Lab3$ cd 3.2
asus@asus-virtual-machine:~/Desktop/Lab3/3.2$ ls
Ex1.c  Ex1.o  Ex1.out  Ex2.c  Ex2.o  Ex2.out
asus@asus-virtual-machine:~/Desktop/Lab3/3.2$ gcc -c Ex1.c
asus@asus-virtual-machine:~/Desktop/Lab3/3.2$ gcc -o Ex1.out Ex1.o
asus@asus-virtual-machine:~/Desktop/Lab3/3.2$ ./Ex1.out 5
Sum of 1 to 5 = 15
Sum of divisor of 5 = 6
asus@asus-virtual-machine:~/Desktop/Lab3/3.2$ ./Ex1.out 6
Sum of 1 to 6 = 21
Sum of divisor of 6 = 12
asus@asus-virtual-machine:~/Desktop/Lab3/3.2$ ./Ex1.out -5
Dot so khong la so nguyen duong
asus@asus-virtual-machine:~/Desktop/Lab3/3.2$ ./Ex1.out ab
Dot so khong la so nguyen duong
asus@asus-virtual-machine:~/Desktop/Lab3/3.2$
```

Bài 2:



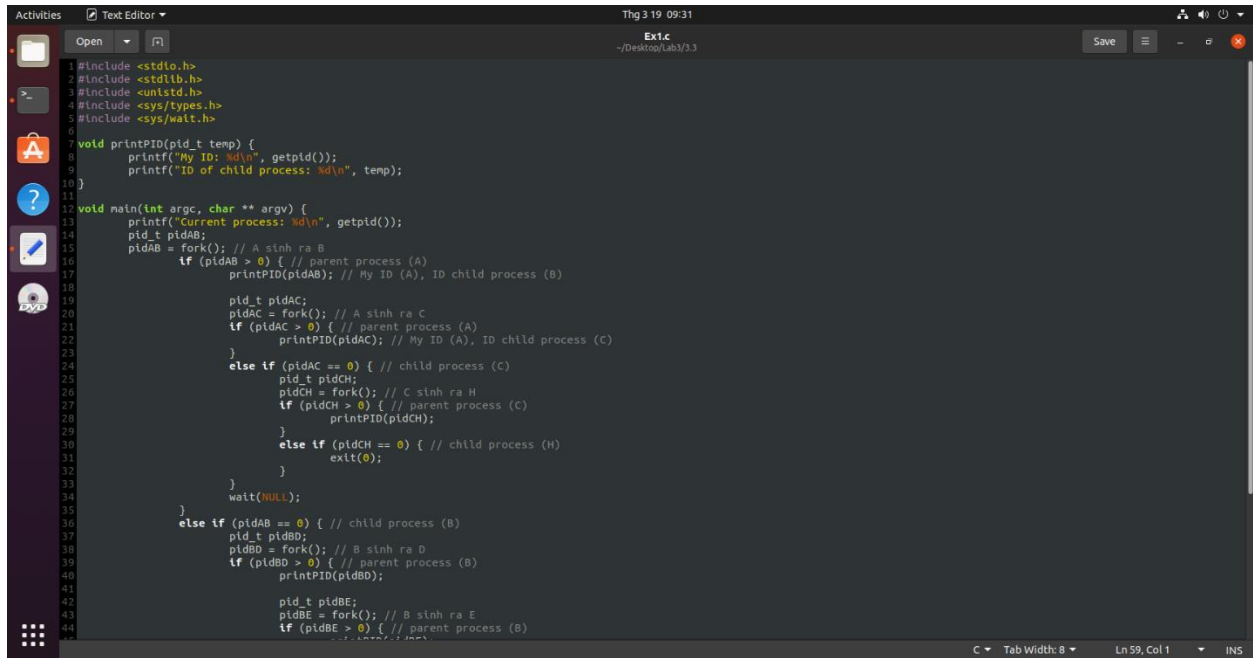
```
1#include <stdio.h>
2#include <stdlib.h>
3#include <unistd.h>
4#include <sys/types.h>
5#include <sys/wait.h>
6
7void collatzSequence (int n) {
8    while (n != 1) {
9        if (n % 2 != 0) {
10            printf("%d, ", n);
11            n = 3*n + 1;
12        }
13        else {
14            printf("%d, ", n);
15            n /= 2;
16        }
17    }
18    printf("%d\n", n);
19}
20
21void main(int argc, char ** argv) {
22    pid_t pid;
23    int n = atoi(argv[1]);
24    if (n <= 0) {
25        printf("Doi so khong la so nguyen duong\n");
26    }
27    else {
28        pid = fork();
29        if (pid == 0) { // child process
30            collatzSequence(n);
31        }
32        else if (pid > 0) { // parent process
33            wait(NULL);
34            printf("Ket thuc tien trinh con\n");
35        }
36    }
37}
```



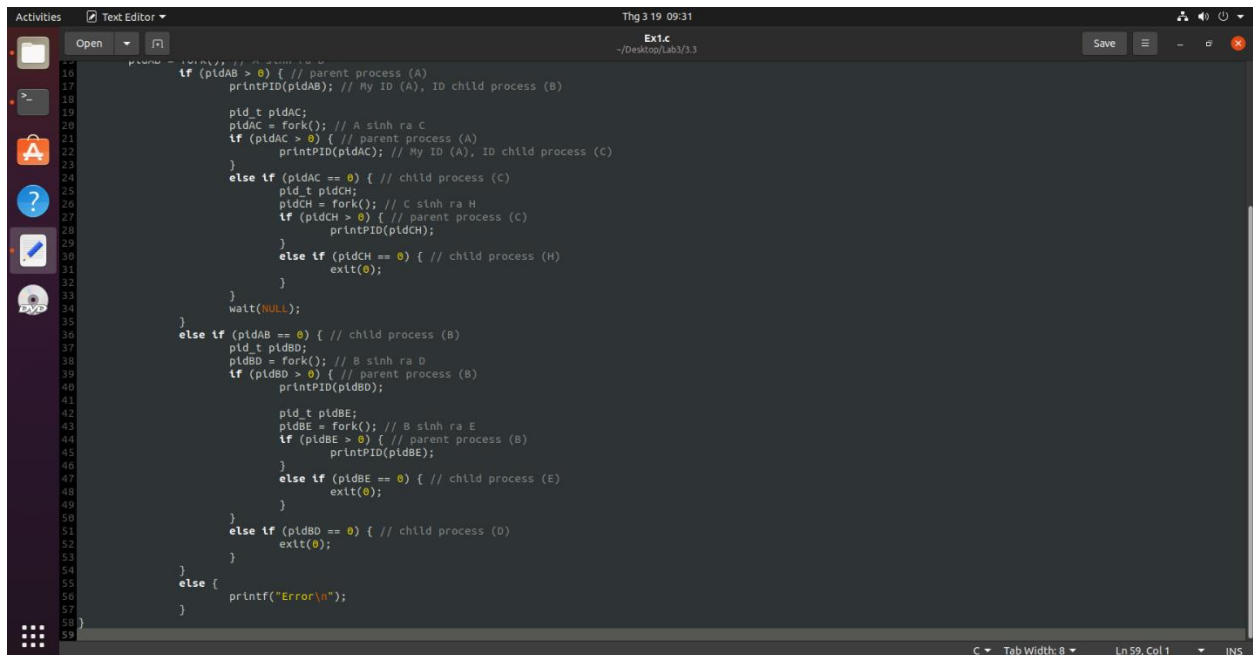
```
asus@asus-virtual-machine: ~/Desktop/Lab3/3.2
asus@asus-virtual-machine:~$ cd Desktop
asus@asus-virtual-machine:~/Desktop$ cd Lab3
asus@asus-virtual-machine:~/Desktop/Lab3$ cd 3.2
asus@asus-virtual-machine:~/Desktop/Lab3/3.2$ gcc -c Ex2.c
asus@asus-virtual-machine:~/Desktop/Lab3/3.2$ gcc -o Ex2.out Ex2.o
asus@asus-virtual-machine:~/Desktop/Lab3/3.2$ ./Ex2.out 35
35, 106, 53, 160, 80, 40, 20, 10, 5, 16, 8, 4, 2, 1
Ket thuc tien trinh con
asus@asus-virtual-machine:~/Desktop/Lab3/3.2$
```

3.3 – Cây tiến trình

Bài 1:



```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <unistd.h>
4 #include <sys/types.h>
5 #include <sys/wait.h>
6
7 void printPID(pid_t temp) {
8     printf("My ID: %d\n", getpid());
9     printf("ID of child process: %d\n", temp);
10 }
11
12 void main(int argc, char ** argv) {
13     printf("Current process: %d\n", getpid());
14     pid_t pidAB;
15     pidAB = fork(); // A sinh ra B
16     if (pidAB > 0) { // parent process (A)
17         printPID(pidAB); // My ID (A), ID child process (B)
18     }
19     pid_t pidAC;
20     pidAC = fork(); // A sinh ra C
21     if (pidAC > 0) { // parent process (A)
22         printPID(pidAC); // My ID (A), ID child process (C)
23     }
24     else if (pidAC == 0) { // child process (C)
25         pid_t pidCH;
26         pidCH = fork(); // C sinh ra H
27         if (pidCH > 0) { // parent process (C)
28             printPID(pidCH);
29         }
30         else if (pidCH == 0) { // child process (H)
31             exit(0);
32         }
33     }
34     wait(NULL);
35 }
36 else if (pidAB == 0) { // child process (B)
37     pid_t pidBD;
38     pidBD = fork(); // B sinh ra D
39     if (pidBD > 0) { // parent process (B)
40         printPID(pidBD);
41     }
42     pid_t pidBE;
43     pidBE = fork(); // B sinh ra E
44     if (pidBE > 0) { // parent process (B)
```

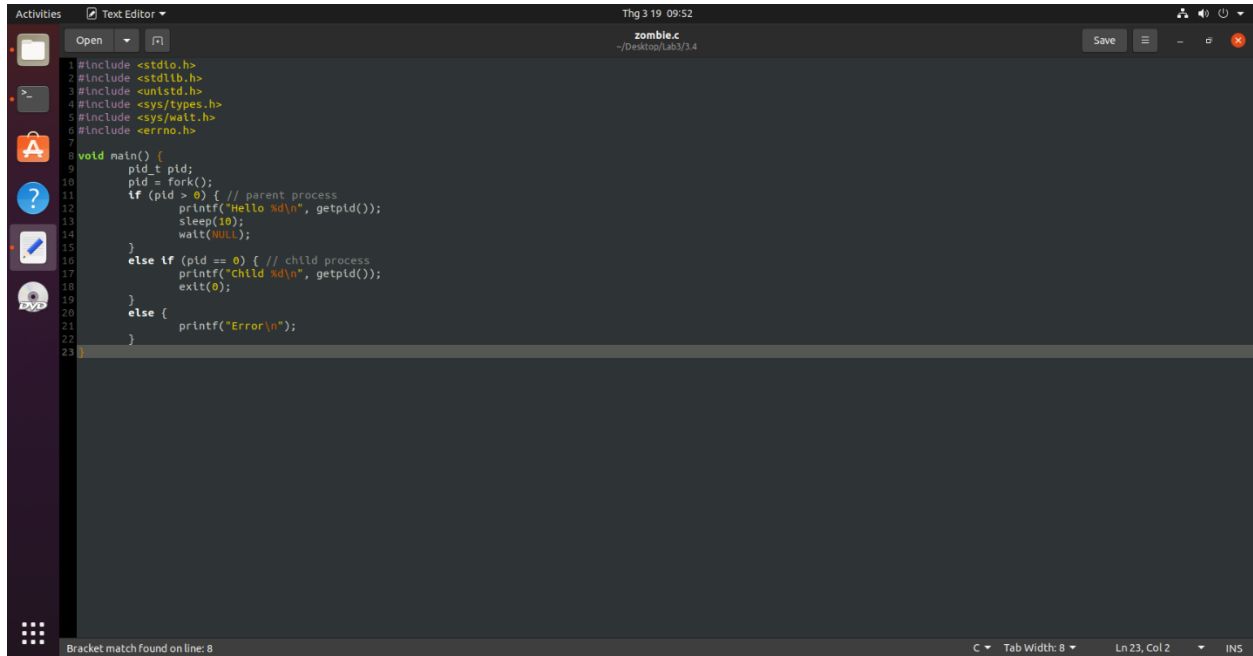


```
45         printPID(pidBE);
46     }
47     else if (pidBE == 0) { // child process (E)
48         exit(0);
49     }
50 }
51 else if (pidBD == 0) { // child process (D)
52     exit(0);
53 }
54 }
55 else {
56     printf("Error\n");
57 }
58 }
59 }
```

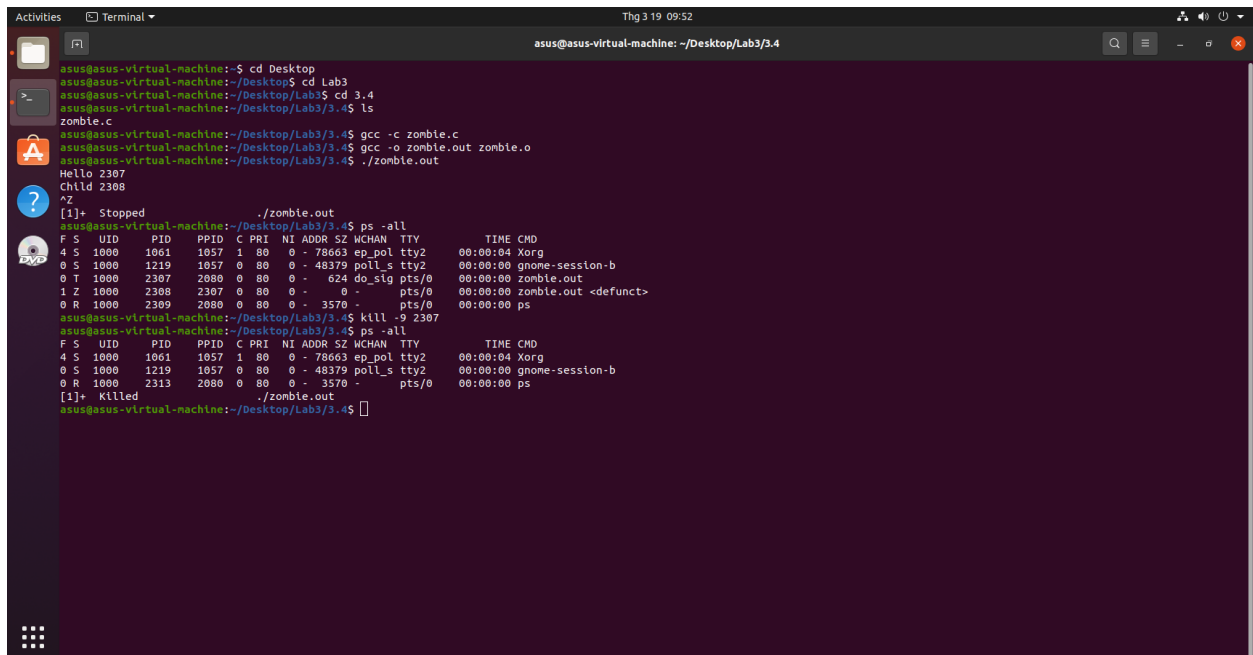
```
Activities Terminal Thg 3 19 09:31
asus@asus-virtual-machine: ~/Desktop/Lab3/3.3
asus@asus-virtual-machine:~$ cd Desktop
asus@asus-virtual-machine:~/Desktop$ cd Lab3
asus@asus-virtual-machine:~/Desktop/Lab3$ ls
3.2 3.3
asus@asus-virtual-machine:~/Desktop/Lab3$ cd 3.3
asus@asus-virtual-machine:~/Desktop/Lab3/3.3$ ls
Ex1.c Ex1.o Ex1.out
asus@asus-virtual-machine:~/Desktop/Lab3/3.3$ gcc -c Ex1.c
asus@asus-virtual-machine:~/Desktop/Lab3/3.3$ gcc -o Ex1.out Ex1.o
asus@asus-virtual-machine:~/Desktop/Lab3/3.3$ ./Ex1.out
Current process: 3963
My ID: 3963
ID of child process: 3964
My ID: 3963
ID of child process: 3965
My ID: 3965
ID of child process: 3966
My ID: 3964
ID of child process: 3967
My ID: 3964
ID of child process: 3968
asus@asus-virtual-machine:~/Desktop/Lab3/3.3$
```

3.4 – Đồng bộ kết thúc các tiến trình

Bài thực hiện và hủy tiến trình zombie

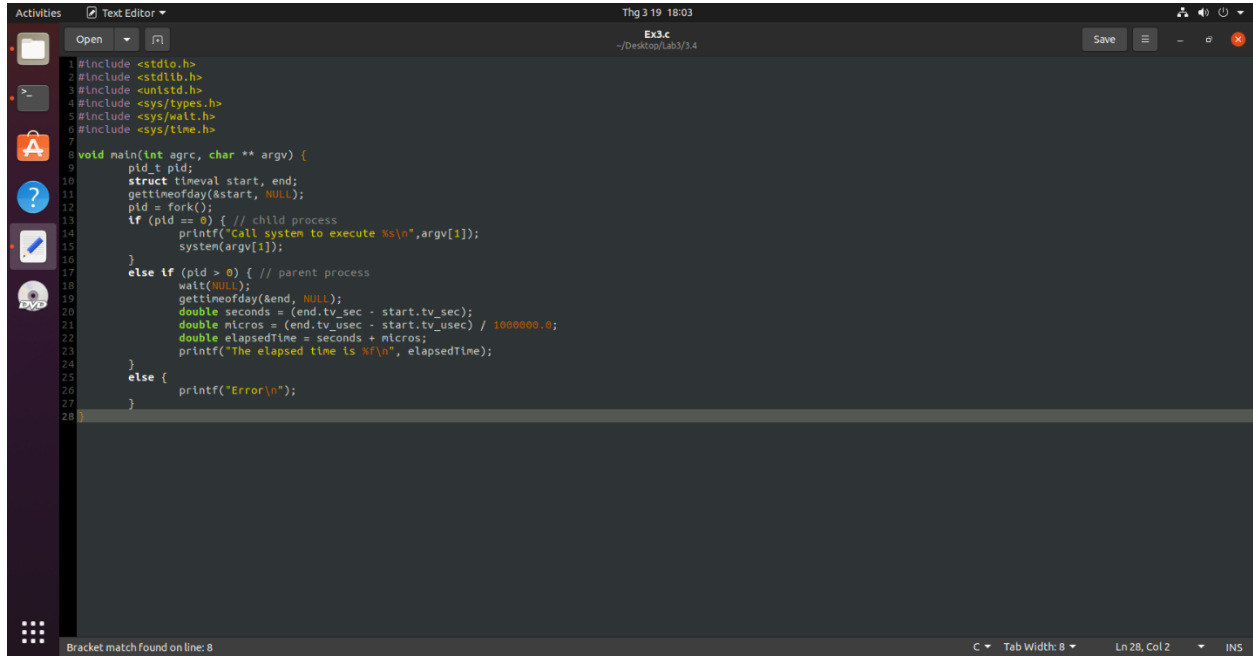


```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <unistd.h>
4 #include <sys/types.h>
5 #include <sys/wait.h>
6 #include <errno.h>
7
8 void main() {
9     pid_t pid;
10    pid = fork();
11    if (pid > 0) { // parent process
12        printf("Hello %d\n", getpid());
13        sleep(10);
14        wait(NULL);
15    }
16    else if (pid == 0) { // child process
17        printf("Child %d\n", getpid());
18        exit(0);
19    }
20    else {
21        printf("Error\n");
22    }
23 }
```



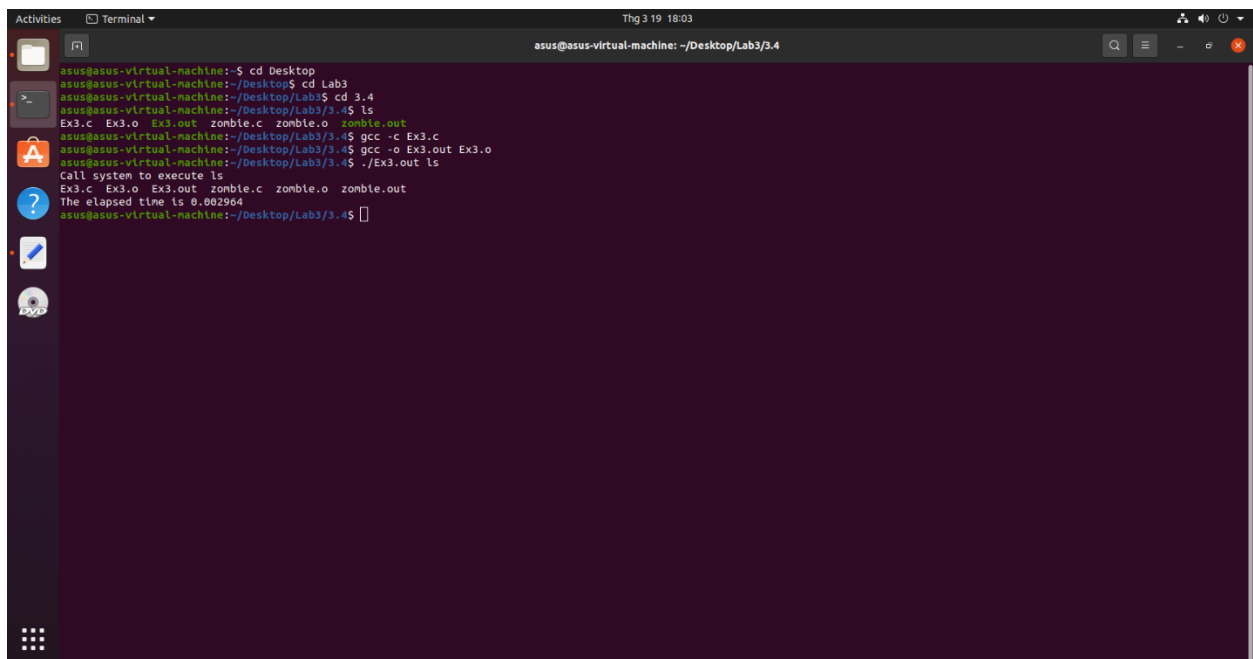
```
asus@asus-virtual-machine: ~/Desktop/Lab3/3.4
asus@asus-virtual-machine:~$ cd Desktop
asus@asus-virtual-machine:~/Desktop$ cd Lab3
asus@asus-virtual-machine:~/Desktop/Lab3$ cd 3.4
asus@asus-virtual-machine:~/Desktop/Lab3/3.4$ ls
zombie.c
asus@asus-virtual-machine:~/Desktop/Lab3/3.4$ gcc -c zombie.c
asus@asus-virtual-machine:~/Desktop/Lab3/3.4$ gcc -o zombie.out zombie.o
asus@asus-virtual-machine:~/Desktop/Lab3/3.4$ ./zombie.out
Hello 2307
Child 2308
^Z
[1]+  Stopped                  ./zombie.out
asus@asus-virtual-machine:~/Desktop/Lab3/3.4$ ps -all
F S UID PID PPID C PRI NI ADDR SZ WCHAN TTY          TIME CMD
4 S 1000 1001 1057 1 80  0 - 78663 ep_pol tty2      00:00:04 Xorg
0 S 1000 1219 1057 0 80  0 - 48379 poll_s tty2      00:00:00 gnome-session-b
0 T 1000 2307 2080 0 80  0 - 624 do_sig pts/0        00:00:00 zombie.out
1 Z 1000 2308 2307 0 80  0 - 0 - pts/0        00:00:00 zombie.out <defunct>
0 R 1000 2309 2080 0 80  0 - 3570 - pts/0        00:00:00 ps
asus@asus-virtual-machine:~/Desktop/Lab3/3.4$ kill -9 2307
asus@asus-virtual-machine:~/Desktop/Lab3/3.4$ ps -all
F S UID PID PPID C PRI NI ADDR SZ WCHAN TTY          TIME CMD
4 S 1000 1001 1057 1 80  0 - 78663 ep_pol tty2      00:00:04 Xorg
0 S 1000 1219 1057 0 80  0 - 48379 poll_s tty2      00:00:00 gnome-session-b
0 R 1000 2313 2080 0 80  0 - 3570 - pts/0        00:00:00 ps
[1]-  Killed                    ./zombie.out
asus@asus-virtual-machine:~/Desktop/Lab3/3.4$
```

Bài 3:



```
1#include <stdio.h>
2#include <stdlib.h>
3#include <unistd.h>
4#include <sys/types.h>
5#include <sys/wait.h>
6#include <sys/time.h>
7
8void main(int argc, char ** argv) {
9    pid_t pid;
10    struct timeval start, end;
11    gettimeofday(&start, NULL);
12    pid = fork();
13    if (pid == 0) { // child process
14        printf("Call system to execute %s\n", argv[1]);
15        system(argv[1]);
16    }
17    else if (pid > 0) { // parent process
18        wait(NULL);
19        gettimeofday(&end, NULL);
20        double seconds = (end.tv_sec - start.tv_sec);
21        double micros = (end.tv_usec - start.tv_usec) / 1000000.0;
22        double elapsedTime = seconds + micros;
23        printf("The elapsed time is %f\n", elapsedTime);
24    }
25    else {
26        printf("Error\n");
27    }
28}
```

Bracket match found on line: 8



```
asus@asus-virtual-machine: ~/Desktop/Lab3/3.4
asus@asus-virtual-machine:~$ cd Desktop
asus@asus-virtual-machine:~/Desktop$ cd Lab3
asus@asus-virtual-machine:~/Desktop/Lab3$ cd 3.4
asus@asus-virtual-machine:~/Desktop/Lab3/3.4$ ls
Ex3.c  Ex3.o  Ex3.out  zombie.c  zombie.o  zombie.out
asus@asus-virtual-machine:~/Desktop/Lab3/3.4$ gcc -c Ex3.c
asus@asus-virtual-machine:~/Desktop/Lab3/3.4$ gcc -o Ex3.out Ex3.o
asus@asus-virtual-machine:~/Desktop/Lab3/3.4$ ./Ex3.out ls
Call system to execute ls
Ex3.c  Ex3.o  Ex3.out  zombie.c  zombie.o  zombie.out
The elapsed time is 0.002964
asus@asus-virtual-machine:~/Desktop/Lab3/3.4$
```